Webデザイン実習4B

2019/11/26

Kazuma Sekiguchi
class@cieds.jp

課題

- スマートフォンに最適化したウェブサイトの構築
  - RWDにする必要は無い（してももちろんOK）
- レストランの紹介および予約サイト
  - 名前や場所などは適当に
  - 画像は商用利用可のもの、または自分で撮影、作成したもののみ許可
  - キャラクターは自分で作成したもの以外使用禁止
- トップページ、紹介部分、予約ページが必要
- 効果的にインタラクションを用いること

課題

- 幅414px 高さ896pxのiPhone11をターゲット
  - Chromeのモバイルモードで表示可能であること
  - 確認はChrome78のモバイルモードで確認します
- jQueryはバージョン3以上のみ使用許可
  - 動かないプラグインが多数出てくるので、注意すること
- 動画の利用はもちろん可
- 各ページにリンクを張ること
  - 1ページサイトもOK。ただし、ページ内リンクは用意すること

- 12月10日の授業時に提出

課題提出物

- HTML
- CSS（HTML埋め込みでもOK）
- JS（HTML埋め込みでもOK）
- 画像ファイル群、動画ファイル群
- デザインファイル（psd ， ai ， sketch ， xd ， figmaなどなど）、バージョンは最新版でOK
  - デザインファイルのPDFファイル

<html5 APIs>

HTML5 API

- HTML5ではJavaScriptを使って初めて利用することのできる機能が大量に盛り込まれている
  - Geolocation（位置情報の取得）
  - ローカルストレージ（ローカル環境のKey-ValueStore）
  - FileAPI（ローカルファイルの取り扱い）
  - WebSocket（P2P通信など）
  - MediaCapture（カメラや映像を取得）

などなど

位置情報へのアクセス(GeolocationAPI)

- HTML5からGPSなどを利用した位置情報へアクセス可能
  - デバイスや回線などによって位置情報に誤差が生じる
  - 誤差も取得可能
    - 一定間隔で取得し、誤差の少ないものを採用する（電池の減りが激しくなるため注意）
    - 最初はめちゃくちゃ誤差があるので、500ミリ秒くらい回して取得するのが一般的
  - 緯度、経度、誤差、高度、高度の誤差、方角、速度を取得可能
    - 一部取得できないものもある

```
navigator.geolocation.getCurrentPosition( success, error );
//取得できたらsuccessを呼び出す。失敗したらerrorを呼び出す
```

取得できるもの

- 関数の引数にeとかを指定しておく
  - このときのe.のeはイベントオブジェクト
  - 緯度:e.coords.latitude
  - 経度:e.coords.longitude
  - 緯度経度の誤差:e.coords.accuracy
  - 高度:e.coords.altitude
  - 高度の誤差:e.coords.altitudeAccuracy
  - 方角:e.coords.heading
  - 速度:e.coords.speed

位置情報へのアクセス

- 取得時にアラートが出るため、ユーザが許可しない限りデータを取得できない
- なぜかPCなどでも取得可能
  - IPアドレス、GPS、無線LAN、携帯基地局などのデータから位置を特定する

位置情報

- 緯度・経度が取得できたところで嬉しくはない
  - 地図上にプロットするなどしないと場所が分からない
  - 地図＝GoogleMap
- GoogleMapは緯度と経度で場所を示すことが可能
  - GoogleMapから地図データを取得し、場所を表示する
- 最近GoogleMapは有料化した
  - 一意のKeyを入れないと警告が出る
  - 一応OKとかをクリックすれば開発モードとして利用可能

GoogleMapへのプロット

- 位置情報で取得できても地図が無いとどこだかはっきりしない
  - GoogleMapを呼び出して、位置を示すことが可能
  - GoogleMapを使うときはAPIKeyの取得が必要
    - Googleのアカウントから取得することが可能
    - 利用回数に応じて課金するための仕組みになっている
    - keyはkey=の形で指定（1行で記述する）

```
<script
src="http://maps.google.com/maps/api/js?key=AIzaSyBHC79-
llHjtupDxzDzavuh5TEWk1Q1DKs"></script>
```

GoogleMapへのプロット

- 位置情報をGoogleMap用の情報に変更
  - オプションを指定

```
var latlng = new google.maps.LatLng(e.coords.latitude, e.coords.longitude);
var myOptions = {
        zoom: 18,//地図の拡大数値
        center: latlng,//地図をどこを中心にして表示するか
        mapTypeId: google.maps.MapTypeId.ROADMAP//どういう地図を表示するか
};
var map = new google.maps.Map(mapcanvas, myOptions);//mapcanvasというDOM要素に表示する指定
var marker = new google.maps.Marker({
        position: latlng,//マーカーをどこに立てるか
        map: map//どの地図に立てるか
});
```

センサーとかJSから利用

- VibrationAPI＝スマートフォン自体の振動を起こすことが可能
  - navigator.vibration(500);//500ms振動
- deviceorientation＝デバイスの傾きを検知

```
window.addEventListener(“deviceorientation”,deviceorientationHandler);
// ジャイロセンサーの値が変化
function deviceorientationHandler(event) {
var beta = event.beta; // X軸
var gamma = event.gamma; // Y軸
var alpha = event.alpha; // Z軸
}
```

スマホセンサー

- 機種によって搭載されているセンサーに違いはある
- HTML5 APIから取得できないものも多い

- GPS
- ジャイロ（傾き）
- カメラ
- 気温
- 湿度
- 気圧
- 圧力計
- 照度計
- 近接センサー
- 加速計

場所とかをデバッグ

- GoogleChromeのDevToolで「More Tools」から「Sensors」を選択すると、スマートフォンで利用可能なさまざまなセンサーの値を変更可能
  - そのときの状態を確認することが可能
  - 結構便利

LocalStorage

- ブラウザー内にkey=valueの形で任意のデータを保存することのできる仕組み
- LocalStorageもJSから利用する
  - HTML5対応ブラウザのみ利用可能な点注意
  - LocalStorageは永久保存（ブラウザから明示的にクリアされたら終わり）
    - 開発者ツールから任意のデータを削除することも可能
  - SessionStorageはブラウザを開いている間だけ有効

```
//localStorageへの保存
localStorage.setItem(“productName",”Xperia");
localStorage.setItem(“productPrice",”65000");

//localStorageからの読み出し
localStorage.getItem(“productName”);//Xperiaが取り出せる
```

LocalStorageの保存

- 基本的にはkey=valueの形で投入する
  - 例えば、「name=“ミク”」という形
  - 同じkey名のものは投入できない（投入すると上書きになる）
  - 複数のデータを投入可能であるが、上記制約があるため、違う名前にする必要がある
  - JSONデータをぶち込む、という乱暴とも言える方法で複数データ（同一のKey名も可）を投入することも可能
  - そのままでは無理なので、JSONデータを文字列に変換する（JSON.stringify()メソッドを利用）
  - データを取り出して利用する時にJSON.parse()メソッドを使うことで、JSONデータに戻すことが可能